RECEIVED
CENTRAL FAX CENTER
OCT 24 2006

REMARKS

Reconsideration and further examination of the present application is respectfully requested. The Office Action has rejected claims 1-20. Applicant has amended claim 1 with this response. No claims were previously canceled. No claims have been added.

I. Claim Rejections – 35 U.S.C. §102

a. Claims 1, 3, and 6 and 9 are rejected under 35 U.S.C. 102 (e) as being anticipated by Marko et al. U.S. Pat. No. 6,876,835 ("**Marko**")

Claims 1, 3, 6, and 9 have been rejected under 35 U.S.C. § 102 (e) as being anticipated by **Marko**. Applicant does not admit that **Marko** is prior art and reserves the right to swear behind the reference at a later date. As discussed below, **Marko** fails to teach or suggest at least one limitation included in each of the claims, and, thus, fails to teach or make obvious the claimed invention.

Overview of Marko

With reference to Fig. 1 of the cited reference, **Marko** essentially shows and describes a system for storing broadcast content files for on-demand playback purposes. The system requires the following components to implement the patented method: a programming center 20; at least one satellite 16; a control center 18; and a system receiver 14.

"The programming center 20 is configured to obtain content from different sources and providers which can comprise both analog and digital information such as audio, video, data, ... and so on." See **Marko** at col. 4, lines 4-8. The "programming center 20 is provided to generate and transmit a composite data stream via the satellites 12 and 16 which comprises a plurality of broadcast channels." See id. at col. 3, lines 52-55. A composite data stream may be "generated,

for example, by time division multiplexing a plurality of broadcast channels, along with other data such as overhead data." See id. at col. 4, lines 56-60. Further, the "file to be transferred via the digital broadcast system such as the system 10 is partitioned at the program center 20 ... for transmission as segments 36, as shown in Figs. 3 and 4." See id. at col. 4, lines 48-54. "The segments 36 are provided with headers 37 to facilitate their capture in a local storage device at receiver 14." See id. at col. 4, lines 57-60. "The partitioning of the file 34 allows for smaller portions of the file (e.g., 8kbps channels) to be interspersed with other broadcast content, hereby reducing the demand on the instantaneous bandwidth of the system." Id. at col. 4, lines 54-57. Each segment of the broadcast file includes an associated segment header which is made up of information such as broadcast ID, auxiliary data, file number, segment number, and total segments. See id. Figs. 4-5.

"The receivers are configured to demultiplex a received composite data stream using synchronization symbols and the slot control field data to playback a selected one of the broadcast channels." See id. at col. 3, line 66 – col. 4, line 4. Additionally, receiver 14 functions to capture and store segments. See id. at Fig. 7.

Finally, control center 18 is provided for telemetry, tracking and control of the satellites. See id. at col. 3, lines 51-52.

Independent claim 1

Regarding independent claim 1, the Office Action states that "**Marko** teaches a computer implemented method for storing data comprising: storing a composite data stream so that it may be restored." [Office Action, 7/27/06, page 2].

Storing the composite data stream where storing includes: "decomposing ...;" "segmenting ...;" and "discarding ... ."

The system in **Marko** "can broadcast a composite data stream 30 generated, for example, by time division multiplexing a plurality of broadcast channels, along with other data such as overhead data." Id. at col. 3, lines 56-60. In **Marko**, the composite data stream consists of the overhead data and a plurality of time slots for transporting traffic channels. See id. at col. 3, lines 61-65. The overhead data is comprised of a slot control field and master synchronization symbol. Id. Each time slot contains a broadcast segment and its associated segment header which typically includes information such as broadcast ID, auxiliary data, file number, segment number, and total segments. See id. at Fig. 5; col. 4, line 64 – col. 5, line 1. So, the composite data stream in **Marko** comprises all of the following: 1) overhead data including master synchronization symbol and slot control field; 2) a plurality of broadcast segments; and 3) the associated segment headers (broadcast ID, file number, etc ...).

However, neither the **Marko** transmitter nor the **Marko** receiver stores the composite data stream where storing includes: "decomposing ...;" "segmenting ...;" and "discarding ... ." Although the transmitter in **Marko** stores the composite data stream in order to transmit the same, the transmitter does not decompose, segment or discard as a part of its storing. Concomitantly, the receiver in **Marko** does not store the composite data stream. The receiver only stores the segments and does not store either the overhead data or the associated segment headers. See id. at col. 6, lines 41-45 ("[a]s shown in FIG. 7, the converter 55 comprises an RF-to-audio converter 54 and an RF-to-control data converter 56 to extract, respectively, the traffic (e.g. the segments 36) and control data (e.g., headers 37) from the received signal"); see also id. at col. 6, lines 51-55 ("[i]n accordance with the present invention, the converter 55 removes the segment headers 37 from the received data stream and determines from the broadcast ID field 38

or the file number field 40 whether the segment is intended for that receiver 14"). The transmitted segment headers are not stored in the **Marko** receiver's local memory. The only function of the segment headers is to facilitate the "capture and storage of segments." See id. at col. 6, lines 8-9.

Additionally, the receiver in **Marko** only stores **one of many** of the segments which are broadcast. The **Marko** receiver only stores those parts of the broadcast (composite) data stream that the **user selects** at the receiver. See id. at col. 2, lines 58-62 ("the receiver is configured to monitor the codes corresponding to the content in the broadcast signal and to capture content segments having one of the codes selected by the user"). The segment header information is compared with the user-selected broadcast ID or file number to determine which segments to store in local memory. See id. at col. 6, lines 58-59 ("[i]f the segment is intended for that receiver, it is stored in the local storage device 50"). Thus, it is only the **user-selected** segments that are stored in local memory and not any other of the broadcast segments. Even if **Marko** could be redesigned to store the entire composite data stream (which consists of the overhead data, the plurality of broadcast segments, and the plurality of associated segment headers) in the receiver's local memory, it still only stores the **one of many** segments of the broadcast data stream selected by the user. One of many is not the total broadcast transmitted, and is not the composite data stream. Since the overhead data, the de-selected plurality of broadcast segments, and the segment headers associated with the plurality of de-selected broadcast segments are not stored in local memory of the receiver in **Marko**, the entirety of the composite data stream is not stored. For this reason, the **Marko** reference does not anticipate claim 1 because neither the **Marko** transmitter nor the **Marko** receiver stores the **composite data stream** where storing includes: "decomposing ...;" "segmenting ...;" and "discarding ... ."

Decomposing composite data stream and segmenting its constituents

The Office Action states that **Marko** teaches "[d]ecomposing the composite data stream into a plurality of constituent data streams," and "segmenting at least one of the plurality of constituent data streams." [Office Action, p. 3].

However, the **Marko** reference does not disclose segmenting the constituent data streams. In **Marko**, once the composite data stream is received at the receiver, it is converted "to extract, respectively, the traffic (e.g., the segments 36) and control data (e.g., headers) from the received signal." See id. at col. 6, lines 40-45. The system disclosed in **Marko** does not perform segmentation on the resulting segments. Instead, the system in **Marko** stores the segments as they were received. See id. at col. 6, lines 58-59 ("[i]f the segment is intended for that receiver, it is stored in the local storage device 50").

In contrast, claim 1 requires decomposing the composite data stream into a plurality of constituent data streams prior to the segmenting operation. Thus, under claim 1, the composite data stream is first decomposed into its constituent data streams, and, only then, is segmentation performed. This is done to increase compression efficiency, and reduce storage consumption. See spec. at p. 18. For example, independent claim 1 recites as currently amended:

1. A computer implemented method for storing data comprising:

storing a composite data stream so that it may be restored, said storing including,

decomposing the composite data stream into a plurality of constituent data streams;

segmenting at least one of the plurality of constituent data streams decomposed from the composite data stream; and

discarding those of the segments resulting from said segmenting which are determined to have been stored previously.

Since the result of decomposing the composite data stream is "a plurality of constituent data streams," and segmenting is performed on constituent data streams; it necessarily follows that a particular composite data stream must be decomposed into its constituent data streams prior to segmenting its constituent data streams. This is performed in this order to facilitate compression efficiency, and reduction of storage consumption. Thus, the amended language, "constituent data streams decomposed from the composite data stream" is non-narrowing because the limitation "decomposed from the composite data stream" was inherent and already present in claim 1.

Therefore, since there is no segmentation performed on the traffic data (segments) in **Marko**, and the traffic data is a constituent of the composite data streams; there necessarily is no segmentation of the constituent data streams as required by claim 1. Thus, **Marko** does not disclose "segmenting at least one of a plurality of constituent data streams" as required by the claim 1.

Accordingly, Applicant submits that **Marko** does not anticipate claim 1 and respectfully requests withdrawal of the claim rejection.

Dependent claims 3 and 6

Claims 3 and 6 are each dependent on claim 1 and incorporate all the limitations contained therein. Therefore, for at least the same reasons as claims 1 is not anticipated by **Marko**, Applicant submits claims 3 and 6 are also not anticipated. Accordingly, Applicant respectfully requests the withdrawal of the claim rejections.

II. Claim Rejections – 35 U.S.C. §103

a. Claim 7 is rejected under 35 U.S.C. 103(a) as being unpatentable over **Marko** in view of Muthiatacharoen (Athicha Muthiatacharoen, Benjie Chen, and David Mazieres "A Low-bandwidth Network File System", MIT Laboratory for Computer Science and NYU Department of Computer Science) ("**Muthiatacharoen**").

Applicant does not admit that **Marko** or **Muthiatacharoen** are prior art and reserves the right to swear behind either reference at a later date.

Independent Claim 7

The Office Action further states, "**Marko** teaches ... storing each of said plurality of composite data streams so that it may be restored." [Office Action at p. 7]. However, **Marko** fails to teach "storing each of said plurality of composite data streams so that it may be restored."

Storing and restoring the composite data stream

The Office Action asserts that the composite data stream is restored when it is captured in a local storage at the receiver. See id. at p. 8 ("See column 4, lines 57-60 "As shown in FIG. 5, the segments are provided with headers to facilitate their capture in a local storage device at the receiver."). However, **Marko** does not have the ability to store and/or restore the composite data stream.

A system that complies with claim 7 has the ability to store and restore the composite data stream. Restoring the composite data stream may be necessary, for example, in a backup system when a user's system crashes and all or part of the local data has been lost. In such a case, the user's system may need the data to be "restored" to the state that existed prior to the system crash. There is nothing in the **Marko** disclosure that performs this function. As discussed above, once the receiver in **Marko** has received, converted and demultiplexed the

composite data stream, it stores only the one segment of the composite data stream that the user has selected. It does not store the **composite data stream**. Since the receiver in **Marko** has no way to retrieve the un-stored parts of the composite data stream (such as the overhead data, the de-selected plurality of broadcast segments, and the segment headers associated with the de-selected plurality of broadcast segments), it cannot restore that which it cannot retrieve. Thus, the receiver in **Marko** cannot restore the composite data stream. Accordingly, Applicant submits that **Marko** fails to teach or suggest "storing each of said plurality of composite data streams so that it may be restored" as required by claim 7.

Moreover, it cannot be argued that the original programming content is that which is "restored" under **Marko**. While it is true that the receiver is capable of storage and playback of the original programming content (the broadcast content that originated from a programming source), such content is just the segment (or traffic) component of the composite data stream and not the composite data stream itself. As discussed above, the composite data stream contains all of the following: 1) overhead data; 2) a plurality of broadcast segments; and 3) the segment headers associated with the plurality of broadcast segments. Storing and playing back the original programming content via the receiver and loudspeakers is not tantamount to "restoring" the composite data stream. Thus, **Marko** fails to teach "storing each of said plurality of composite data streams so that it may be restored." Accordingly, Applicant respectfully submits that claim 7 is not anticipated by **Marko** and respectfully requests withdrawal of the claim rejection.

Decomposing composite data stream and segmenting its constituents

Additionally, the Office Action states, "**Marko** teaches "decomposing the composite data stream into a plurality of constituent data streams," and "segmenting the constituent data stream." See id. However, in accordance with the above discussion as it pertains to claim 1,

Marko fails to teach or suggest "decomposing the composite data stream into a plurality of constituent data streams," and "segmenting the constituent data stream."

Accordingly, Applicant submits that claim 7 is not anticipated by the disclosure in **Marko** for the same reasons as claim 1 is not anticipated. Therefore, Applicant respectfully requests withdrawal of the claim rejection.

The Office Action states that "claim 7 is ... unpatentable over **Marko** in view of **Muthiatacharoen**." See id. at p. 7. The Office Action further states that **Muthiatacharoen** teaches "storing using segment reuse a set of one or more of said plurality of constituent data streams ...," and "[i]t would have been obvious to one with ordinary skill in the art at the time of the invention to combine the teachings of **Marko** with **Muthiatacharoen** because of the efficiency and storage space savings benefits that are provided by using segment reuse." See id. at p. 9.

A person having ordinary skill in the art would not have combined **Marko** and **Muthiatacharoen** because the systems are so different that there would have been no likelihood of success

A person having ordinary skill in the art would not have combined **Marko** and **Muthiatacharoen** because the systems are so different that there would have been no likelihood of success. The system disclosed in **Marko** and **Muthiatacharoen** are directed to two completely different systems. The system disclosed in **Marko** is a broadcast system that picks segments out of a broadcast data stream and stores them for on-demand playback. However, the system in **Muthiatacharoen** is about reducing bandwidth in a network file storage system. **Muthiatacharoen** contemplates a client-server system that utilizes segment reuse to take advantage of commonality between files on a file storage network. In **Muthiatacharoen**, traffic across the network can be reduced when a client checks out a file from the server because in such

a system there is a lot of commonality between files. When a particular client checks out a file, the server only sends the segments of that file across the network that have been changed since the client last checked the file out. Thus, the **Muthiatacharoen** system cross-checks the segments of files stored on the server with those segments in the client's cache memory to increase bandwidth efficiency in a network file storage system. See **Muthiatacharoen** at p. 5, sec. 3.2.1.

This is different from **Marko** because in **Marko** the receiver is storing segments of broadcast content. There would be a much reduced expectation of commonality between segments stored previously and those currently being captured because content of a broadcast transmission is inherently changing from one broadcast to the next. As disclosed in **Marko**, "[t]he type of content which can be distributed in an SDARS system or a similar digital broadcast system typical [sic] includes audio programs such as music recordings, news programs and talk shows, among other programs, and advertisements." See **Marko** at col. 1, lines 48-53. People generally listen to talk shows and news broadcasts, not because the content is the same from one time to the next, but because the content changes. It is expected that the **Marko** system will receive different content from one time to the next. Thus, a person having ordinary skill in the art would not have combined the references as suggested by the Office Action since such a person would recognize that a system in **Muthiatacharoen** would have a lot of commonality between files and a system in **Marko** would not.

Therefore, a person of ordinary skill in the art would not have combined **Marko** with **Muthiatacharoen** because the systems are so different that there would have been no likelihood of success. For this reason, Applicant submits that claim 7 is not unpatentable over **Marko** in view of **Muthiatacharoen** and respectfully requests withdrawal of the claim rejection.

Even if **Marko** and **Muthiatacharoen** are combinable, a person having ordinary skill in the art would not have combined them in the way the Office Action has.

Even if **Marko** and **Muthiatacharoen** are combinable, a person having ordinary skill in the art would not have combined them in the way the Office Action has. **Muthiatacharoen** discloses a network file system that saves bandwidth by taking advantage of commonality between files. See **Muthiatacharoen** at p. 13, second column. As such, **Muthiatacharoen** requires a client-server network. See id. at p. 5, Sec. 3.2.1 ("[w]hen a user opens a file, if the file is not in the local cache or the cached version is not up to date, the client fetches a new version from the server").

The Office Action basically pulls the segment reuse portion of the **Muthiatacharoen** system and places it into **Marko**'s storage space "because of the efficiency and storage space saving benefits that are provided by segment reuse." [Office Action, p. 9]. However, since **Muthiatacharoen** is about bandwidth reduction in a networked file storage system, and **Marko** has bandwidth transmission; a person of ordinary skill in the art, when faced with these two references, would apply the **Muthiatacharoen** system on the bandwidth transmission of **Marko**, and not on the storage space. To do this would require **Marko** to include a two-way communication system so that the receiver could request only those segments that are updated from what was already in the receiver's local storage.

However, **Marko** does not teach or suggest a client-server relationship between the receiver and the transmitter (programming center). There is no way for programming center to receive data from the receiver indicating the status or content of the receiver's local memory. The receiver disclosed in **Marko** does not transmit data streams back to the programming center. In fact, the transmitter contemplated in **Marko** broadcasts composite data streams without regard to the local memory content of the receiver. As a result, there is no way for the system disclosed

in **Marko** to take advantage of the commonality between files as contemplated by **Muthiatacharoen**. What would result would be something quite different from what the Office Action asserts.

Therefore, even if **Marko** and **Muthiatacharoen** are combinable, a person having ordinary skill in the art would not have combined them in the way the Office Action has. Accordingly, Applicant submits that claim 7 is not unpatentable over **Marko** in view of **Muthiatacharoen** and respectfully requests withdrawal of the claim rejection.

Even if **Marko** and **Muthiatacharoen** are combined in the way the Office Action asserts, the combination would not read on claim 7.

Even if **Marko** and **Muthiatacharoen** are combined in the storage space as asserted by the Office Action, this combination would not read on claim 7. In this case, the combination of **Marko** and **Muthiatacharoen** would result in a system that captures broadcast segments and stores them using segment reuse. However, the thing that such a system would store would be the broadcast segment selected by the user, and that is not what Applicant is claiming with claim 7. According to the above discussion with regard to claim 1, **Marko** in not capable of storing the composite data stream which is what claim 7 is claiming. So, combining **Marko** and **Muthiatacharoen** would not result in a system that stores the composite data stream as required by claim 7. Accordingly, Applicant submits that claim 7 is not unpatentable over **Marko** in view of **Muthiatacharoen** and respectfully requests withdrawal of the claim rejection.

The exemplary advantages of claim 7 render a system that complies with claim 7 nonobvious

Finally, the Office Action states, "[i]t would have been obvious to one with ordinary skill in the art at the time of the invention to combine the teachings of **Marko** with **Muthiatacharoen** because of the efficiency and storage space savings benefits that are provided by using segment reuse." Id. However, the advantages of a system that complies with claim 7 is an indication that

the limitations of claim 7 are nonobvious over the cited references. By way of example, existing systems that generate composite data streams and segment them for storage typically segment the composite data stream. See spec. at Fig. 7A. As a result, "storage servers that store entire composite data streams are relatively inefficient in that they store large amounts of redundant data." See spec. p. 4, paragraph 0006. In contrast, a system that complies with claim 7, in one exemplary embodiment, is useful for storing a composite data stream more efficiently through decomposing the composite data stream first, and then applying segment reuse to the constituent data streams. An exemplary use that complies with claim 7, therefore, removes the administrative data from such a composite data stream before segmenting the constituent data streams so as to enable better segment matching. This allows segmenting to work better as illustrated in spec. at Fig. 7B. While the claim is not limited to the system shown in Fig. 7B, it is capable of enabling it.

Therefore, for this additional reason, claim 7 is nonobvious over the cited references. As a result, the combination fails to render claim 7 obvious as contended by the Office Action. Accordingly, Applicant submits that claim 7 is not unpatentable over **Marko** in view of **Muthiatacharoen** and respectfully requests withdrawal of the claim rejection.

Dependent claim 9

Claim 9 is dependent on claim 7 and incorporates all the limitations contained therein. Therefore, for at least the same reasons as claims 7 is not anticipated by **Marko**, Applicant submits claim 9 also is not anticipated. Accordingly, Applicant respectfully requests the withdrawal of the claim rejection.

b. Claims 2, 8, 13, and 16 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Marko** in view of USENIX – "Proceedings of the FAST 2002 Conference and File and Storage Technologies", Monterey, California, January 28-30, 2002 ("**USENIX**").

Applicant does not admit that **Marko** or **USENIX** are prior art and reserves the right to swear behind either reference at a later date.

Dependent Claim 2 and 8

Claims 2 and 8 are each dependent on one of claims 1 and 7, either directly or indirectly, and incorporate all the limitations contained therein. Therefore, for at least the same reasons as claims 1 and 7 are also not anticipated by **Marko**, Applicant submits claims 2 and 8 are not anticipated. Accordingly, Applicant respectfully requests the withdrawal of the claim rejections.

c. Claims 4, 5, 10, and 11 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Marko** in view of Hattrup et al., (App. No. 10/447,617) ("**Hattrup**")

Applicant does not admit that **Marko** or **Hattrup** are prior art and reserves the right to swear behind either reference at a later date.

Dependent Claim 4, 5, and 10

Claim 4, 5, 10 and 11 are each dependent on claim 1, either directly or indirectly, and incorporate all the limitations contained therein. Therefore, for at least the same reasons as claim 1 is not anticipated by **Marko**, Applicant submits claims 4, 5, 10, and 11 are also not anticipated. Accordingly, Applicant respectfully requests the withdrawal of the claim rejections.

d. Claims 12 and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Marko** in view of **Muthiatacharoen**, and in further view of **Hattrup**.

Independent Claim 12

The Office Action states that claim 12 is unpatentable over **Marko** in view of **Muthiatacharoen**, and in "further view of **Hattrup**," [Office Action, p. 10].

Storing each of said plurality of composite data streams so that it may be restored

The Office Action further states that **Marko** teaches "a computer implemented method for storing data comprising: storing a composite data streams so that it may be restored." Id. However, in accordance with the above argument as it pertains to claims 1 and 7, **Marko** fails to teach or suggest "storing each of said plurality of composite data streams so that it may be restored." Accordingly, Applicant submits that claim 12 is not anticipated by the disclosure in **Marko** for the same reasons as claims 1 and 7 are not anticipated. Therefore, Applicant respectfully requests withdrawal of the claim rejection.

Decomposing composite data stream and segmenting its constituents

Additionally, the Office Action states, "**Marko** teaches "decomposing the composite data stream into a plurality of constituent data streams," and "segmenting the constituent data stream." Id. at p. 8. However, in accordance with the above argument as it pertains to claim 1, **Marko** fails to teach or suggest "decomposing the composite data stream into a plurality of constituent data streams," and "segmenting the constituent data stream." Accordingly, Applicant submits that claim 12 is not anticipated by the disclosure in **Marko** for the same reasons as claim 1 is not anticipated. Therefore, Applicant respectfully requests withdrawal of the claim rejection.

Additionally, the Office Action states that claim 12 is "unpatentable over **Marko** ... in view of **Muthiatacharoen** ... and in further view of **Hattrup**." [Office Action, pp. 9-10].

Even if the combination of **Marko**, **Muthiatacharoen** and **Hattrup** would result in a system that is capable of "backing up each of said plurality of constituent data streams separately applying segment reuse," the combination would not read on claim 12

The Office Action further states that **Hattrup** teaches "backing up each of said plurality of constituent data streams separately," and "[i]t would have been obvious to one with ordinary skill in the art at the time of the invention to combine the teachings of **Marko** with the teachings of **Hattrup** because, as stated in **Hattrup**, backing up the streams separately provide for additionally [sic] privacy and security for data." See id. at pp. 10-11. However, according to the above discussion, **Marko** only stores that the user-selected segment of the composite data stream. Accordingly, the remaining parts or "constituents" of the transmitted composite data stream (including the overhead data, the plurality of de-selected broadcast segments, and the segment headers associated with the plurality of de-selected broadcast segments) are not stored. The system in **Marko** cannot retrieve the un-stored parts of the composite data stream. Thus, it is impossible for the system disclosed in **Marko** to backup that which was not originally stored. Therefore, **Hattrup** cannot backup "***each*** of said plurality of constituent data streams."

Likewise, according to the above discussion with regard to claim 7, claim 12 is not unpatentable over **Marko** in view of **Muthiatacharoen** for the same reasons as claim 7 is not. Therefore, Applicant submits that claim 12 is not unpatentable over **Marko** in view of **Muthiatacharoen**, and in further view of **Hattrup**. Accordingly, Applicant respectfully requests withdrawal of the claim rejection.

Dependent Claim 13 and 14

Claim 13 and 14 are each dependent, either directly or indirectly, on claim 12 and incorporate all the limitations contained therein. Therefore, for at least the same reasons as claim 12 is not unpatentable over **Marko** in view of the cited references, Applicant submits claims 13

and 14 are also not anticipated. Accordingly, Applicant respectfully requests the withdrawal of the claim rejection.

e. Claims 15 and 19 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Marko** in view of **USENIX** and in further view of **Muthiatacharoen**

Independent Claim 15

The Office Action states that apparatus claim 15 is "unpatentable over **Marko** in view of **USENIX** ... and further in view of **Muthiatacharoen**." See id. at pp. 12-13.

The combination of **Marko** and **USENIX** would not result in "a composite data stream decomposer/recomposer ... to recompose composite data streams" as required by claim 15

The Office action further states that **Marko** teaches "a composite data stream decomposer/recomposer ... to decompose composite data streams into their constituent data streams." Id. at p. 13. However, **Marko** clearly does not disclose a decomposer/recomposer apparatus as required by claim 15. There is no mention of the words "recomposer" in the **Marko** disclosure at all. Accordingly, Applicant submits that **Marko** does not teach or suggest this limitation as required by claim 15.

The Office action states that **USENIX** teaches "recompose composite data streams from their constituent data streams ... ." See id. at p. 13. Thus, the Office Action contends that because **USENIX** teaches "recompose composite data streams from their constituent data streams," it somehow reads on a "recomposer" as required by claim 15. However, claim 15 is an apparatus claim directed to a "recomposer," which is an apparatus not a method. Thus, a method to "recompose composite data streams from their constituent data streams" would not read on an "recomposer" apparatus. A "recomposer" apparatus is not disclosed or even mentioned in the

USENIX reference. Therefore, **Marko** does not disclose a "recomposer" as required by claim 15, and **USENIX** fails to cure this deficiency.

Furthermore, according to the above discussion pertaining to claims 1 and 7, the system in **Marko** is unable to store the composite data stream. It necessarily follows, then, that the system in **Marko** is unable to recompose that which it is unable to store. The **Marko** receiver cannot recompose the composite data stream because it failed to store several parts including the overhead data, a plurality of de-selected broadcast segments, and the segment headers associated with the plurality of de-selected broadcast segments. Even if **USENIX** teaches "recompose composite data streams from their constituent data streams," it cannot recompose what **Marko** failed to store. Thus, the combination of **Marko** and **USENIX** would not result in "a composite data stream decomposer/recomposer ... to recompose composite data streams" as required by claim 15.

Likewise, according to the above discussion with regard to claim 7, claim 12 is not unpatentable over **Marko** in view of **Muthiatacharoen** for the same reasons as claim 7 is not unpatentable over **Marko** in view of **Muthiatacharoen**. Therefore, Applicant submits that claim 15 is not unpatentable over **Marko** in view of **USENIX** and in further view of **Muthiatacharoen**. Accordingly, Applicant respectfully requests withdrawal of the claim rejection.

Dependent Claims 16, 17, 18, 19 and 20

Claims 16, 17, 18, 19 and 20 are each dependent on claim 15, either directly or indirectly, and incorporate all the limitations contained therein. Therefore, for at least the same reasons as claim 15 is not unpatentable over **Marko** in view of the cited references, Applicant submits that

claims 16, 17, 18, 19 and 20 are also not unpatentable. Accordingly, Applicant respectfully requests the withdrawal of the claim rejections.

CONCLUSION

Applicant respectfully submits that the rejections have been overcome by the amendments and remarks, and that the claims as amended are now in condition for allowance. Accordingly, Applicant respectfully requests the rejections be withdrawn and the claims as amended be allowed.

Invitation for a telephone interview

If the Examiner believes a telephone conference would expedite or assist in the allowance of the present application, the Examiner is invited to call the undersigned attorney at (408) 962-7581.

Charge our Deposit Account

Please charge any shortage to our Deposit Account No. 02-2666.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: October 24, 2006

Matthew W. Hindman
Reg. No. 57, 396
Matthew_Hindman@bstz.com

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, California 90025-1026
(408) 720-8300